**TRƯỜNG ĐẠI HỌC VĂN LANG**

**KHOA CÔNG NGHỆ THÔNG TIN**

**A red and black shield with white logo

AI-generated content may be incorrect.**

**BÁO CÁO ĐỒ ÁN MÔN HỌC**

[**THỊ**](https://elearning.vlu.edu.vn/course/view.php?id=7239) **GIÁC MÁY TÍNH**

***Đề tài:***

**NHẬN DIỆN NGÔN NGỮ KÝ HIỆU**

**BẰNG YOLO11**

**SVTH:**

**Nguyễn Văn Phú – 224802010661**

**Thái Viết Lập -2274802010**

**Nguyễn Công Tiến - 2274802010**

**GVHD: TS. Đỗ Hữu Quân**

***Hồ Chí Minh-Tháng 3 năm 2025***

# **LỜI CẢM ƠN**

Chúng em xin gửi lời cảm ơn chân thành đến **Trường Đại học Văn Lang** đã tạo điều kiện thuận lợi để chúng em được học tập và nghiên cứu trong học phần *Thị giác máy tính*. Môn học đã cung cấp cho chúng em nhiều kiến thức hữu ích, giúp mở rộng hiểu biết và khả năng ứng dụng trí tuệ nhân tạo (AI) vào thực tế, đặc biệt trong việc nhận dạng và phân loại đối tượng.

Đặc biệt, chúng em xin bày tỏ lòng biết ơn sâu sắc đến **Thầy Đỗ Hữu Quân** – người đã tận tâm giảng dạy, hướng dẫn và chia sẻ những kiến thức quý báu trong suốt quá trình thực hiện đồ án. Sự tận tụy và nhiệt huyết của thầy đã giúp chúng em hiểu rõ hơn về các thuật toán xử lý hình ảnh, cũng như ứng dụng mô hình **YOLO** trong việc nhận diện ngôn ngữ ký hiệu Việt Nam. Những bài giảng sinh động cùng sự hướng dẫn sát sao của thầy không chỉ nâng cao kiến thức chuyên môn mà còn truyền cảm hứng để chúng em tiếp tục nghiên cứu và phát triển các sản phẩm ứng dụng thị giác máy tính trong thực tiễn.

Dù đã cố gắng hết sức để hoàn thành đồ án, song do hạn chế về kinh nghiệm và kiến thức thực tế, chắc chắn vẫn còn nhiều thiếu sót. Chúng em rất mong nhận được những góp ý quý báu từ thầy để có thể hoàn thiện đề tài tốt hơn trong tương lai.

**Chúng em xin chân thành cảm ơn!**

**Mục Lục**

[**LỜI CẢM ƠN** 2](#_Toc194161811)

[**CHƯƠNG 1: GIỚI THIỆU DỰ ÁN** 5](#_Toc194161812)

[**1.1** **. Lý do chọn đề tài** 5](#_Toc194161813)

[**1.2** **. Mục tiêu đề tài** 6](#_Toc194161814)

[**1.3** **. Phạm vi dự án** 7](#_Toc194161815)

[**1.4** **. Kết chương** 7](#_Toc194161816)

[**CHƯƠNG 2: CƠ SỞ LÝ THUYẾT** 8](#_Toc194161817)

[**2.1** . **Tìm hiểu về thị giác máy tính và nhận diện ký hiệu ngôn ngữ** 8](#_Toc194161818)

[**2.1.1** *Giới thiệu về thị giác máy tính* 8](#_Toc194161819)

[**2.1.2** *Nhận diện ngôn ngữ ký hiệu* 9](#_Toc194161820)

[**2.2** **. Giới thiệu về YOLO** 11](#_Toc194161821)

[**2.2.1** *. Tổng quan về YOLO* 11](#_Toc194161822)

[**2.2.2** *.Kiến trúc YOLO11* 12](#_Toc194161823)

[**2.3** **. Các phương pháp nhận diện ngôn ngữ ký hiệu** 13](#_Toc194161824)

[*2.3.1* *. Nhận Diện Ngôn Ngữ Ký Hiệu Dựa Trên Hình Ảnh Tĩnh* 13](#_Toc194161825)

[*2.3.2* *. Nhận Diện Ngôn Ngữ Ký Hiệu Dựa Trên Video Thời Gian Thực* 14](#_Toc194161826)

[**2.4** **. Sử dụng OpenCV trong nhận diện thời gian thực** 14](#_Toc194161827)

[**2.5** **. Kết chương** 16](#_Toc194161828)

[**CHƯƠNG 3: QUY TRÌNH THỰC HIỆN** 17](#_Toc194161829)

[**3.1** **. Dữ liệu** 17](#_Toc194161830)

[**3.2** **. Huấn luyện mô hình** 17](#_Toc194161831)

[**3.3** **. Triển khai mô hình YOLO11 trong nhận diện SL** 17](#_Toc194161832)

[**3.4** **. Cài đặt giao diện người dùng với Streamlit** 18](#_Toc194161833)

[**3.5** **. Kết chương** 18](#_Toc194161834)

[**CHƯƠNG 4: KẾT QUẢ VÀ ĐÁNH GIÁ** 19](#_Toc194161835)

[**4.1** **. Kết quả nhận diện SL từ Webcam** 19](#_Toc194161836)

[**4.2** **. Đánh giá hiệu suất mô hình** 32](#_Toc194161837)

[**4.3** **. Kết chương** 34](#_Toc194161838)

[**CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN** 35](#_Toc194161839)

[**5.1** **. Kết luận** 35](#_Toc194161840)

[**5.2** **. Hướng phát triển** 35](#_Toc194161841)

[**TÀI LIỆU THAM KHẢO** 37](#_Toc194161842)

**Mục Lục Hình Ảnh**

[Hình 1. Hình ảnh minh họa Computer Vision 9](#_Toc194161843)

[Hình 2. Hình ảnh minh họa YOLO nhận diện vật thể. 11](#_Toc194161844)

[Hình 3. Hình ảnh minh họa kiến trúc YOLO11. 12](#_Toc194161845)

[Hình 4. Hình ảnh Confusion Matrix. 20](#_Toc194161846)

[Hình 5. Hình ảnh Confusion Matrix Normalized. 21](#_Toc194161847)

[Hình 6.Hình ảnh F1-Confidence Curve. 22](#_Toc194161848)

[Hình 7. Hình ảnh Precision-Recall Curve. 23](#_Toc194161849)

[Hình 8. Hình ảnh Recall-Confidence Curve. 24](#_Toc194161850)

[Hình 9. Hình ảnh Training Loss. 25](#_Toc194161851)

[Hình 10. Hình ảnh Validation Loss. 26](#_Toc194161852)

[Hình 11. Hình ảnh Metrics. 29](#_Toc194161853)

[Hình 12. Hình ảnh tương quan. 30](#_Toc194161854)

[Hình 13. Hình ảnh giao diện streamlit khi thực realtime. 32](#_Toc194161855)

# **CHƯƠNG 1: GIỚI THIỆU DỰ ÁN**

## **. Lý do chọn đề tài**

Việc phát triển và ứng dụng công nghệ nhận dạng hình ảnh vào đời sống thực tế đã trở thành xu hướng nổi bật trong vài năm tới. Trong đó, nhận dạng ngôn ngữ ký hiệu là một trong những ứng dụng quan trọng hỗ trợ giao tiếp giữa người điếc và cộng đồng. Ngôn ngữ ký hiệu, đặc biệt là SL (Ngôn ngữ ký hiệu Mỹ), được sử dụng rộng rãi trong cộng đồng người khiếm thính nhưng vẫn còn nhiều trở ngại đối với những người không hiểu ngôn ngữ ký hiệu khi giao tiếp với người khiếm thính. Điều này dẫn đến nhu cầu cấp thiết là phát triển các công cụ hỗ trợ nhận dạng ngôn ngữ ký hiệu tự động, chính xác và dễ sử dụng.

Một trong những vấn đề giao tiếp chính của người khiếm thính là họ không thể tương tác dễ dàng với những người không hiểu ngôn ngữ ký hiệu. Các giải pháp hiện tại, chẳng hạn như dịch trực tiếp từ người hiểu ngôn ngữ ký hiệu, tuy hiệu quả nhưng lại gặp phải một số khó khăn, chẳng hạn như phải dựa vào sự có mặt của người hiểu ngôn ngữ ký hiệu và không phải lúc nào cũng tạo điều kiện thuận lợi cho giao tiếp theo thời gian thực. Vì vậy, việc phát triển hệ thống nhận dạng cử chỉ SL tự động và trực tiếp sẽ mang lại nhiều lợi ích cho cộng đồng người khiếm thính, đặc biệt giúp họ giao tiếp dễ dàng hơn trong cuộc sống hàng ngày.

Hệ thống nhận dạng ngôn ngữ ký hiệu SL sử dụng công nghệ học sâu như YOLO11 có thể giải quyết vấn đề này bằng cách nhận dạng nhanh chóng và chính xác cử chỉ từ video hoặc hình ảnh theo thời gian thực. Mô hình YOLO (Bạn chỉ nhìn một lần) đã được chứng minh là rất hiệu quả trong việc xác định các đối tượng trong ảnh và video, nhanh chóng và có độ chính xác cao, lý tưởng để xác định các cử chỉ động trong SL. Với khả năng nhận dạng đối tượng nhanh chóng và hiệu quả, YOLO11 sẽ là công cụ lý tưởng để phát triển hệ thống nhận dạng ngôn ngữ ký hiệu.

Ngoài ra, việc kết hợp với công nghệ Streamlit giúp xây dựng giao diện người dùng đơn giản, trực quan, dễ sử dụng, đồng thời hỗ trợ hiển thị kết quả nhận dạng trực tiếp từ webcam. Giao diện người dùng dễ sử dụng này sẽ đóng vai trò là cầu nối quan trọng giữa công nghệ và người dùng, đặc biệt đối với những người chưa biết nhiều về công nghệ.

Một lý do khác để chọn đề tài này là sự phát triển mạnh mẽ của công nghệ nhận dạng hình ảnh trong nhiều lĩnh vực, đặc biệt là y tế, giáo dục và truyền thông. Việc ứng dụng công nghệ nhận dạng ngôn ngữ ký hiệu không chỉ giúp ích cho người khiếm thính mà còn mang đến cơ hội nghiên cứu và phát triển sâu hơn các ứng dụng khác như hỗ trợ học tập cho người khiếm thính, hỗ trợ giao tiếp với khách hàng và những người chưa quen với ngôn ngữ ký hiệu.

Cuối cùng, việc chọn đề tài này phản ánh một xu hướng ngày càng phổ biến, đó là việc kết hợp công nghệ vào các vấn đề xã hội. Đây thực sự là một bước tiến lớn, khi công nghệ không chỉ giúp nâng cao chất lượng cuộc sống của các cộng đồng yếu thế, mà còn mở ra nhiều cơ hội mới để giải quyết những vấn đề thực tế mà xã hội đang đối mặt. Cụ thể, dự án này không chỉ giúp người khiếm thính hòa nhập tốt hơn với cộng đồng mà còn thúc đẩy việc áp dụng các công nghệ tiên tiến vào các lĩnh vực có ảnh hưởng sâu rộng, mang lại những giá trị lâu dài và bền vững.

Với những lý do đó, việc nghiên cứu và phát triển hệ thống nhận diện ngôn ngữ ký hiệu SL bằng YOLO11 không chỉ là một hướng đi hợp lý mà còn là một đóng góp có ý nghĩa, không chỉ trong công nghệ mà còn trong việc cải thiện cuộc sống của cộng đồng người khiếm thính.

## **. Mục tiêu đề tài**

Mục tiêu chính của đề tài này là phát triển một hệ thống nhận diện ngôn ngữ ký hiệu SL( Sign Language ) tự động, sử dụng mô hình YOLO11 kết hợp với giao diện người dùng Streamlit. Hệ thống này sẽ giúp nhận diện các ký hiệu tay trong thời gian thực qua video webcam, phục vụ cho việc giao tiếp giữa người khiếm thính và người không biết ngôn ngữ ký hiệu. Các mục tiêu cụ thể bao gồm:

* **Xây Dựng Mô Hình Nhận Diện SL**: Sử dụng YOLO11 để huấn luyện và xây dựng mô hình nhận diện các ký hiệu tay trong ngôn ngữ SL. Mô hình sẽ được huấn luyện với bộ dữ liệu các ký hiệu SL để đảm bảo khả năng nhận diện chính xác các ký hiệu trong môi trường thực tế.
* **Triển Khai Nhận Diện Thời Gian Thực**: Phát triển hệ thống nhận diện ký hiệu SL trong thời gian thực, với khả năng xử lý video webcam và nhận diện các ký hiệu tay ngay lập tức. Điều này sẽ giúp tạo ra một công cụ giao tiếp tiện lợi cho người khiếm thính và cộng đồng.
* **Tạo Giao Diện Người Dùng Dễ Dàng Sử Dụng**: Sử dụng Streamlit để xây dựng giao diện người dùng đơn giản và trực quan, cho phép người dùng dễ dàng bật tắt webcam, xem kết quả nhận diện ký hiệu và tương tác với hệ thống mà không gặp khó khăn.
* **Đánh Giá Hiệu Suất Mô Hình**: Đánh giá độ chính xác và hiệu suất của mô hình YOLO11 trong việc nhận diện các ký hiệu SL trong các điều kiện khác nhau. Đặc biệt, tập trung vào các yếu tố như độ chính xác của mô hình, thời gian xử lý và khả năng làm việc trong môi trường thực tế.
* **Nâng Cao Khả Năng Hòa Nhập Xã Hội Cho Người Khiếm Thính**: Hướng đến việc phát triển một công cụ có thể giúp người khiếm thính giao tiếp một cách tự nhiên và dễ dàng hơn với cộng đồng, qua đó nâng cao sự hòa nhập xã hội và giảm bớt những khó khăn trong việc truyền đạt thông tin.

## **. Phạm vi dự án**

Đề tài này tập trung vào việc xây dựng và triển khai một hệ thống nhận diện ngôn ngữ ký hiệu SL( Sign Language ) sử dụng công nghệ YOLO11 kết hợp với giao diện người dùng Streamlit. Mục đích chính của hệ thống là nhận diện các ký hiệu tay trong ngôn ngữ SL từ video webcam trong thời gian thực, giúp người khiếm thính và cộng đồng giao tiếp dễ dàng hơn.

Các hoạt động chính của nghiên cứu bao gồm:

* **Thu thập và chuẩn bị dữ liệu SL**: Xây dựng bộ dữ liệu ký hiệu tay SL để huấn luyện mô hình YOLO11.
* **Huấn luyện mô hình YOLO11**: Áp dụng mô hình YOLO11 để nhận diện các ký hiệu tay trong SL từ video webcam.
* **Xây dựng giao diện người dùng**: Sử dụng Streamlit để tạo giao diện đơn giản cho người dùng, giúp họ tương tác với hệ thống nhận diện.
* **Đánh giá hiệu suất mô hình**: Kiểm tra độ chính xác và hiệu quả của mô hình trong các tình huống thực tế.

Mục tiêu chính của nghiên cứu là huấn luyện mô hình YOLO11 để nhận diện chính xác các ký hiệu tay trong SL từ video. Mô hình sẽ được thử nghiệm và đánh giá hiệu suất dựa trên các chỉ số như độ chính xác và tốc độ nhận diện. Ngoài ra, nghiên cứu cũng sẽ phát triển giao diện người dùng dễ sử dụng để hiển thị kết quả nhận diện, giúp người khiếm thính và những người không biết ngôn ngữ ký hiệu có thể giao tiếp một cách dễ dàng và mượt mà.

## **. Kết chương**

Sau khi tìm hiểu chương này, chúng ta đã cùng nhau xem lại các mục tiêu và phạm vi nghiên cứu của đề tài, đồng thời làm rõ những lý do quan trọng để chọn đề tài phát triển hệ thống nhận diện ngôn ngữ ký hiệu SL. Mục tiêu của chúng ta là xây dựng một hệ thống sử dụng công nghệ nhận diện hình ảnh YOLO11 để giúp người khiếm thính giao tiếp hiệu quả hơn với cộng đồng, và giao diện người dùng được phát triển bằng Streamlit sẽ làm cho hệ thống trở nên dễ dàng tiếp cận hơn.

Ngoài ra, chúng ta cũng đã xác định rõ giới hạn của nghiên cứu, từ việc chỉ tập trung vào SL cho đến những thách thức mà hệ thống có thể gặp phải trong môi trường thực tế. Tuy nhiên, những giới hạn này không làm giảm đi giá trị của đề tài, mà ngược lại, nó sẽ là cơ sở để cải thiện và mở rộng hệ thống trong tương lai.

Cuối cùng, mục tiêu chung của không chỉ là tạo ra một công cụ nhận diện ngôn ngữ ký hiệu, mà còn góp phần làm tăng cường sự hòa nhập xã hội cho người khiếm thính. Với những nền tảng đã được xây dựng trong chương này, chúng ta sẽ tiếp tục triển khai các bước tiếp theo để hoàn thiện hệ thống, đánh giá hiệu suất và hoàn thiện các tính năng trong các chương tiếp theo của đồ án.

# **CHƯƠNG 2: CƠ SỞ LÝ THUYẾT**

## . **Tìm hiểu về thị giác máy tính và nhận diện ký hiệu ngôn ngữ**

### *Giới thiệu về thị giác máy tính*

**Thị giác máy tính** là một lĩnh vực của trí tuệ nhân tạo (AI) và học máy (Machine Learning), tập trung vào việc giúp máy tính "nhìn" và hiểu các hình ảnh, video giống như con người. Mục tiêu chính của thị giác máy tính là phát triển các thuật toán và mô hình máy tính để nhận diện, phân tích, và xử lý hình ảnh từ thế giới thực. Các thuật toán này giúp máy tính không chỉ nhận diện đối tượng mà còn đưa ra các quyết định dựa trên thông tin từ các hình ảnh hoặc video.

Các yếu tố cơ bản trong thị giác máy tính bao gồm:

* **Nhận diện đối tượng**: Xác định và phân loại các đối tượng trong hình ảnh hoặc video, ví dụ như nhận diện người, xe cộ, động vật, v.v.
* **Phát hiện và theo dõi đối tượng**: Phát hiện các đối tượng di chuyển trong video và theo dõi chúng qua các khung hình liên tiếp.
* **Phân tích hình ảnh**: Phân tích các đặc trưng hình học, màu sắc và kết cấu trong hình ảnh để rút ra thông tin hữu ích, ví dụ như phân đoạn hình ảnh (image segmentation).
* **Nhận diện khuôn mặt**: Xác định và nhận diện khuôn mặt trong ảnh hoặc video.
* **Hiểu ngữ nghĩa**: Sử dụng các mô hình học sâu để hiểu ngữ nghĩa của hình ảnh, ví dụ như nhận diện các tình huống, hành động hoặc cảm xúc của người trong ảnh.

A computer vision with icons

AI-generated content may be incorrect.

Hình 1. Hình ảnh minh họa Computer Vision

### *Nhận diện ngôn ngữ ký hiệu*

#### . Giới thiệu

**Nhận diện ngôn ngữ ký hiệu** là một ứng dụng đặc biệt của thị giác máy tính, giúp nhận diện và hiểu các cử chỉ tay hoặc biểu hiện của người sử dụng ngôn ngữ ký hiệu. Ngôn ngữ ký hiệu, đặc biệt là **SL( Sign Language )**, được sử dụng rộng rãi trong cộng đồng người khiếm thính để giao tiếp. Tuy nhiên, không phải ai cũng hiểu ngôn ngữ ký hiệu, vì vậy việc phát triển các hệ thống tự động để nhận diện và dịch ngôn ngữ ký hiệu thành văn bản hoặc lời nói là rất quan trọng.

Các hệ thống nhận diện ngôn ngữ ký hiệu sử dụng các kỹ thuật của thị giác máy tính để phân tích các cử chỉ tay và chuyển chúng thành các ký hiệu có thể hiểu được. Quy trình này có thể bao gồm:

* **Phát hiện cử chỉ tay**: Nhận diện các cử chỉ tay trong không gian 2D hoặc 3D, từ đó phân loại cử chỉ tương ứng với các ký hiệu trong ngôn ngữ ký hiệu.
* **Xử lý video thời gian thực**: Dùng các thuật toán thị giác máy tính để phân tích video và nhận diện cử chỉ tay ngay lập tức, tạo ra các hệ thống giao tiếp hiệu quả giữa người khiếm thính và người không biết ngôn ngữ ký hiệu.
* **Sử dụng học sâu (Deep Learning)**: Các mô hình học sâu như YOLO11 hoặc mạng nơ-ron sâu (CNNs) được huấn luyện để nhận diện và phân loại các cử chỉ tay trong ngữ cảnh ngôn ngữ ký hiệu.

#### . Quy Trình Nhận Diện Ngôn Ngữ Ký Hiệu

Quy trình nhận diện ngôn ngữ ký hiệu sử dụng các kỹ thuật của thị giác máy tính để phân tích và nhận diện các cử chỉ tay của người sử dụng ngôn ngữ ký hiệu, sau đó chuyển đổi chúng thành ngữ nghĩa có thể hiểu được (dưới dạng văn bản hoặc âm thanh). Dưới đây là các bước chính trong quy trình này:

* **Tiền xử lý hình ảnh/video**: Trước khi nhận diện, các hình ảnh hoặc video từ webcam được xử lý để loại bỏ nhiễu, cải thiện độ rõ nét và chuẩn bị cho bước tiếp theo. Các kỹ thuật như làm mượt ảnh, phân đoạn (image segmentation), hoặc lọc màu sắc có thể được sử dụng để làm sạch dữ liệu đầu vào.
* **Phát hiện cử chỉ tay**: Sau khi xử lý hình ảnh, bước tiếp theo là phát hiện cử chỉ tay trong không gian 2D hoặc 3D. Các mô hình nhận diện đối tượng, như YOLO11, sẽ giúp phát hiện tay và phân loại cử chỉ. Các kỹ thuật nhận diện đối tượng này có thể sử dụng các mạng nơ-ron sâu để nhận diện các hình dạng và chuyển động của tay.
* **Phân loại cử chỉ tay**: Một khi tay được phát hiện, hệ thống sẽ phân loại cử chỉ tay theo từng ký hiệu trong ngôn ngữ ký hiệu (ví dụ, SL). Điều này có thể được thực hiện thông qua các mô hình học sâu như **mạng nơ-ron tích chập (CNNs)** hoặc các mô hình học máy khác để phân loại cử chỉ thành các ký hiệu cụ thể.
* **Chuyển đổi ký hiệu thành ngữ nghĩa**: Sau khi nhận diện được cử chỉ tay, hệ thống sẽ chuyển đổi các ký hiệu này thành ngữ nghĩa có thể hiểu được, có thể là văn bản hoặc âm thanh. Đây là một phần quan trọng của quy trình nhận diện ngôn ngữ ký hiệu, giúp người sử dụng giao tiếp với người không biết ngôn ngữ ký hiệu.
* **Hiển thị kết quả và giao tiếp**: Cuối cùng, kết quả nhận diện sẽ được hiển thị trên giao diện người dùng (thông qua Streamlit hoặc ứng dụng tương tự), giúp người khiếm thính hoặc người không biết ngôn ngữ ký hiệu có thể nhận ra thông điệp. Các hệ thống cũng có thể tích hợp chức năng chuyển đổi thành văn bản hoặc lời nói để người không biết ngôn ngữ ký hiệu có thể hiểu.

## **. Giới thiệu về YOLO**

### *. Tổng quan về YOLO*

A road with cars and a bus

AI-generated content may be incorrect.

Hình 2. Hình ảnh minh họa YOLO nhận diện vật thể.

* **YOLO** (You Only Look Once) là một mô hình nhận diện đối tượng trong hình ảnh và video, nổi bật với khả năng xử lý nhanh và chính xác. Khác với các mô hình truyền thống, YOLO nhận diện tất cả các đối tượng trong một ảnh hoặc video chỉ trong một lần duy nhất, giúp tiết kiệm thời gian và tài nguyên tính toán.
* **Cách YOLO Hoạt Động:** YOLO chia ảnh thành các ô lưới và mỗi ô dự đoán các đối tượng và vị trí của chúng, cùng với điểm tin cậy (confidence score) và xác suất của từng lớp đối tượng. YOLO sử dụng một mạng nơ-ron tích chập (CNN) duy nhất để phân tích toàn bộ ảnh và trả về kết quả ngay lập tức.
* **Ưu Điểm:**
* **Tốc độ cao**: YOLO xử lý rất nhanh, thích hợp cho các ứng dụng thời gian thực.
* **Độ chính xác cao**: Mặc dù tốc độ nhanh, YOLO vẫn duy trì độ chính xác cao trong nhận diện.
* **Nhược Điểm**
* **Nhận diện đối tượng nhỏ**: YOLO có thể gặp khó khăn trong việc nhận diện đối tượng nhỏ hoặc chồng chéo.
* **Ứng Dụng:** YOLO được sử dụng rộng rãi trong các lĩnh vực như an ninh, xe tự lái, y tế, và thương mại, nhờ vào khả năng nhận diện đối tượng trong thời gian thực và độ chính xác cao.
* **Các Phiên Bản:** YOLO đã có nhiều phiên bản cải tiến, từ YOLOv1 đến YOLO11, với mỗi phiên bản cải thiện về độ chính xác và tốc độ, đặc biệt là trong việc nhận diện các đối tượng nhỏ và xử lý trong môi trường thực tế.

### *.Kiến trúc YOLO11*

YOLO11 là phiên bản mới nhất của dòng mô hình YOLO (You Only Look Once) nổi bật với khả năng nhận diện đối tượng trong thời gian thực. Cấu trúc của YOLO11 có sự cải tiến đáng kể so với các phiên bản trước đó, mang lại sự kết hợp giữa tốc độ và độ chính xác. Dưới đây là tổng quan về kiến trúc của YOLO11:

A diagram of a process

AI-generated content may be incorrect.

Hình 3. Hình ảnh minh họa kiến trúc YOLO11.

YOLO11 tiếp tục kế thừa nguyên lý hoạt động của các phiên bản YOLO trước, đó là chia ảnh thành các ô lưới (grid cells) và dự đoán các đối tượng trong từng ô lưới. Tuy nhiên, YOLO11 đã cải tiến nhiều về mặt mạng nơ-ron, giúp cải thiện khả năng nhận diện đối tượng, đặc biệt là các đối tượng nhỏ, trong khi vẫn duy trì tốc độ xử lý cực nhanh.

Kiến trúc của YOLO11 chủ yếu bao gồm các thành phần sau:

* Backbone: Đây là phần mạng nơ-ron được sử dụng để trích xuất đặc trưng từ ảnh đầu vào. YOLO11 sử dụng các cải tiến từ CSPDarknet (một phần trong các phiên bản trước) hoặc các cấu trúc mạng mạnh mẽ khác để cải thiện khả năng trích xuất thông tin.
* Neck: Phần này có nhiệm vụ kết nối backbone với head, bao gồm các mạng phụ để giúp mô hình nhận diện các đặc trưng ở các mức độ khác nhau của ảnh. YOLO11 sử dụng PANet (Path Aggregation Network) để nâng cao khả năng nhận diện các đối tượng ở các độ phân giải khác nhau.
* Head: Đây là phần cuối cùng của mô hình, có nhiệm vụ dự đoán các hộp giới hạn (bounding boxes), xác suất của các đối tượng và lớp đối tượng trong ảnh. YOLO11 cải thiện việc dự đoán này với các thuật toán mới để tăng độ chính xác và tốc độ.

Các Cải Tiến trong YOLO11:

* Mạng nơ-ron nâng cao: Các cải tiến trong kiến trúc mạng giúp YOLO11 nhận diện các đối tượng nhỏ và các đối tượng trong môi trường phức tạp tốt hơn so với các phiên bản trước.
* Cải tiến trong việc huấn luyện: YOLO11 sử dụng các kỹ thuật huấn luyện mới như augmentation, mixup, và các chiến lược học sâu để tăng độ chính xác mà không làm giảm tốc độ nhận diện.
* Tối ưu hóa hiệu suất: YOLO11 được tối ưu hóa để hoạt động hiệu quả trên các phần cứng khác nhau, từ CPU đến GPU, giúp giảm thiểu thời gian xử lý mà vẫn duy trì độ chính xác cao.

Ứng Dụng của YOLO11:

* Nhận diện đối tượng trong video: YOLO11 có thể nhận diện các đối tượng trong video thời gian thực, được sử dụng trong các hệ thống giám sát an ninh.
* Xe tự lái: YOLO11 được ứng dụng trong việc nhận diện các vật thể như người đi bộ, phương tiện giao thông, và các chướng ngại vật cho xe tự lái.
* Ứng dụng y tế: YOLO11 có thể phân tích các hình ảnh y tế như X-quang hoặc MRI để phát hiện các dấu hiệu bệnh tật.

## **. Các phương pháp nhận diện ngôn ngữ ký hiệu**

Nhận diện ngôn ngữ ký hiệu là quá trình chuyển đổi các cử chỉ tay hoặc biểu cảm khuôn mặt trong ngôn ngữ ký hiệu thành ngữ nghĩa có thể hiểu được (thường là văn bản hoặc âm thanh). Có nhiều phương pháp khác nhau được sử dụng để nhận diện ngôn ngữ ký hiệu, mỗi phương pháp có các ưu nhược điểm và ứng dụng riêng. Dưới đây là các phương pháp phổ biến:

### *. Nhận Diện Ngôn Ngữ Ký Hiệu Dựa Trên Hình Ảnh Tĩnh*

* Mô tả: Phương pháp này sử dụng các kỹ thuật xử lý ảnh để nhận diện các cử chỉ tay từ hình ảnh tĩnh.
* Kỹ thuật sử dụng:
* Trích xuất đặc trưng: Sử dụng các phương pháp như HOG (Histogram of Oriented Gradients) hoặc SIFT (Scale Invariant Feature Transform) để trích xuất các đặc trưng từ hình ảnh.
* Mạng nơ-ron tích chập (CNN): CNN là một phương pháp mạnh mẽ trong nhận diện hình ảnh, giúp nhận diện các cử chỉ tay với độ chính xác cao. CNN có thể học được các đặc trưng hình học và ngữ nghĩa của các cử chỉ tay.
* Ưu điểm:
* Phương pháp này có thể nhận diện các cử chỉ tay trong môi trường ổn định với ít nhiễu.
* Nhược điểm:
* Khó khăn trong việc nhận diện khi có sự thay đổi về ánh sáng, góc nhìn hoặc nền hình ảnh.

### *. Nhận Diện Ngôn Ngữ Ký Hiệu Dựa Trên Video Thời Gian Thực*

Mô tả: Phương pháp này sử dụng mô hình YOLO11 để nhận diện các cử chỉ tay trong ngôn ngữ ký hiệu SL thông qua webcam và hiển thị kết quả trong thời gian thực.

Kỹ thuật sử dụng:

* Chuẩn bị dữ liệu: Dữ liệu huấn luyện bao gồm các hình ảnh cử chỉ tay trong ngôn ngữ ký hiệu SL được tổ chức dưới định dạng YAML, chứa thông tin về các lớp ký hiệu và đường dẫn tới dữ liệu hình ảnh.
* Huấn luyện mô hình YOLO11: Mô hình YOLO11 được huấn luyện trên bộ dữ liệu này với các tham số như kích thước ảnh (1024x1024 pixels), số lượng batch (64), và số epoch (300), không sử dụng pretrained model mà huấn luyện từ đầu.
* Webcam và Streamlit: Giao diện người dùng được xây dựng bằng Streamlit, kết hợp với webcam để nhận diện cử chỉ tay trong thời gian thực.
* Nhận diện và hiển thị kết quả: Mô hình YOLO11 nhận diện các cử chỉ tay, vẽ hộp bao quanh các đối tượng và phân loại các ký hiệu. Các kết quả này được hiển thị ngay trên giao diện Streamlit.

Ưu điểm:

* Thời gian thực: Quá trình nhận diện diễn ra ngay lập tức với tốc độ xử lý nhanh.
* Độ chính xác cao: Mô hình YOLO11 cho kết quả chính xác, nhận diện đúng các cử chỉ tay trong môi trường thay đổi.
* Giao diện đơn giản: Người dùng dễ dàng tương tác và nhận kết quả trực quan qua giao diện Streamlit.

Nhược điểm:

* Yêu cầu phần cứng mạnh: Quá trình huấn luyện và nhận diện yêu cầu phần cứng mạnh, đặc biệt là GPU để xử lý video và huấn luyện mô hình hiệu quả.
* Chuyển đổi ánh sáng: Cần điều chỉnh các tham số ánh sáng trong các môi trường ánh sáng yếu để đảm bảo độ chính xác.

## **. Sử dụng OpenCV trong nhận diện thời gian thực**

OpenCV (Open Source Computer Vision Library) là một thư viện mã nguồn mở mạnh mẽ, cung cấp các công cụ và thuật toán để xử lý hình ảnh và video trong thời gian thực. Nó hỗ trợ nhiều tính năng từ nhận diện khuôn mặt, phát hiện chuyển động đến nhận diện đối tượng trong video. OpenCV là một công cụ quan trọng trong việc xây dựng các hệ thống nhận diện thời gian thực, đặc biệt là trong các ứng dụng như nhận diện ngôn ngữ ký hiệu.

Các Tính Năng của OpenCV trong Nhận Diện Thời Gian Thực

* Nhận Diện Đối Tượng và Phát Hiện Chuyển Động: OpenCV cung cấp các thuật toán mạnh mẽ để nhận diện đối tượng trong các video trực tiếp. Các kỹ thuật như Haar Cascades và Optical Flow có thể được sử dụng để nhận diện và theo dõi các đối tượng trong video thời gian thực.
* Xử Lý Hình Ảnh và Video: OpenCV hỗ trợ các công cụ xử lý hình ảnh cơ bản như lọc, chuyển đổi màu sắc, phân đoạn ảnh (image segmentation) và phân tích các đặc trưng hình học. Điều này cho phép phát hiện và nhận diện các cử chỉ tay hoặc khuôn mặt trong video một cách hiệu quả.
* Tạo Giao Diện Người Dùng: Thư viện OpenCV có thể được kết hợp với các công cụ như Streamlit để xây dựng giao diện người dùng dễ dàng, giúp hiển thị kết quả nhận diện ngay lập tức. Điều này rất hữu ích khi triển khai các ứng dụng nhận diện ngôn ngữ ký hiệu trong thời gian thực, cho phép người sử dụng nhận kết quả ngay trên màn hình.
* Tối Ưu Hóa và Tăng Cường Hiệu Suất: OpenCV giúp tối ưu hóa quá trình xử lý video bằng cách sử dụng các kỹ thuật như multithreading để xử lý song song nhiều tác vụ, từ đó giảm thiểu độ trễ và tăng hiệu quả trong các hệ thống nhận diện thời gian thực.

Ưu Điểm của OpenCV trong Nhận Diện Thời Gian Thực

* Hiệu suất cao: OpenCV được tối ưu hóa cho việc xử lý video và hình ảnh trong thời gian thực, có thể xử lý hàng nghìn khung hình mỗi giây.
* Hỗ trợ đa nền tảng: OpenCV có thể chạy trên nhiều hệ điều hành như Windows, Linux và macOS, đồng thời hỗ trợ nhiều ngôn ngữ lập trình như Python, C++, Java.
* Dễ sử dụng: OpenCV cung cấp một API đơn giản và dễ tiếp cận, giúp lập trình viên nhanh chóng xây dựng và triển khai các ứng dụng nhận diện.

Nhược Điểm của OpenCV trong Nhận Diện Thời Gian Thực

* Khó khăn trong việc nhận diện các đối tượng nhỏ: Mặc dù OpenCV rất mạnh mẽ, nhưng đôi khi việc nhận diện các đối tượng nhỏ trong video có thể gặp khó khăn nếu không có sự kết hợp với các mô hình học sâu.
* Cần phần cứng mạnh: Để đạt được hiệu suất tối đa, các hệ thống nhận diện sử dụng OpenCV yêu cầu phần cứng mạnh mẽ, đặc biệt khi xử lý video có độ phân giải cao.

## **. Kết chương**

Qua chương này chúng ta biết được các phương pháp và công nghệ sử dụng trong việc nhận diện ngôn ngữ ký hiệu, với trọng tâm là mô hình YOLO11 và các công cụ hỗ trợ như OpenCV. Chúng ta đã tìm hiểu quy trình huấn luyện mô hình YOLO11, từ việc chuẩn bị dữ liệu cho đến việc tối ưu hóa và triển khai mô hình trong các ứng dụng nhận diện ngôn ngữ ký hiệu thời gian thực.

Tuy nhiên, còn một số thách thức trong việc nhận diện chính xác các cử chỉ tay trong môi trường thay đổi, như ánh sáng yếu hoặc nền phức tạp. Các cải tiến và tối ưu hóa trong quá trình huấn luyện và triển khai mô hình sẽ giúp nâng cao độ chính xác và tốc độ nhận diện, mang lại hiệu quả cao hơn trong các ứng dụng thực tế.

Với những nền tảng đã được xây dựng trong chương này, việc triển khai mô hình nhận diện ngôn ngữ ký hiệu sẽ tiếp tục được phát triển trong các chương tiếp theo, với mục tiêu hoàn thiện hệ thống và cải thiện khả năng tương tác giữa người khiếm thính và cộng đồng.

# **CHƯƠNG 3: QUY TRÌNH THỰC HIỆN**

## **. Dữ liệu**

Thu thập dữ liệu: Dữ liệu được thu thập từ nhiều nguồn khác nhau, bao gồm các video và hình ảnh của người thực hiện các cử chỉ tay trong ngôn ngữ ký hiệu SL( Sign Language ). Bộ dữ liệu này có thể bao gồm các cử chỉ tay thực hiện trong nhiều điều kiện ánh sáng khác nhau và từ các góc độ khác nhau để mô phỏng các tình huống thực tế.

Tiền xử lý dữ liệu: Trước khi sử dụng dữ liệu cho huấn luyện, các hình ảnh cần được tiền xử lý như:

* Chuyển đổi kích thước: Các hình ảnh đầu vào được chuyển đổi về kích thước chuẩn (ví dụ: 1024x1024 pixels) để mô hình YOLO11 có thể xử lý hiệu quả.
* Chuẩn hóa ảnh: Các giá trị pixel được chuẩn hóa để phù hợp với yêu cầu của mô hình, đảm bảo rằng mô hình có thể học và nhận diện chính xác.
* Augmentation (Tăng cường dữ liệu): Để tăng tính đa dạng và giúp mô hình học tốt hơn, các kỹ thuật tăng cường dữ liệu như xoay ảnh, thay đổi độ sáng, thay đổi góc nhìn, v.v., được áp dụng. Điều này giúp mô hình không bị quá khớp (overfitting) vào các dữ liệu cụ thể.

## **. Huấn luyện mô hình**

Mô hình YOLO11 được huấn luyện để nhận diện và phân loại các cử chỉ tay trong ngữ cảnh ngôn ngữ ký hiệu. Các tham số huấn luyện bao gồm:

* Kích thước ảnh: 1024x1024 pixels, để đảm bảo mô hình có thể xử lý và nhận diện chính xác các cử chỉ tay.
* Batch size: 64, cho phép huấn luyện mô hình với số lượng hình ảnh lớn.
* Số epoch: 100, đảm bảo mô hình học đủ từ bộ dữ liệu.
* Không sử dụng pretrained model: Mô hình YOLO11 được huấn luyện từ đầu, không sử dụng mô hình đã huấn luyện trước đó để phù hợp với bộ dữ liệu ngôn ngữ ký hiệu SL.

Quá trình huấn luyện sử dụng GPU và các kỹ thuật học sâu để tối ưu hóa mô hình. Các tham số như learning rate và patience được điều chỉnh trong quá trình huấn luyện để tránh overfitting.

## **. Triển khai mô hình YOLO11 trong nhận diện SL**

Sau khi mô hình được huấn luyện thành công, bước tiếp theo là triển khai mô hình YOLO11 để nhận diện các cử chỉ tay trong ngôn ngữ ký hiệu SL. Mô hình sẽ sử dụng webcam để nhận diện cử chỉ tay trong thời gian thực.

* Xử lý hình ảnh từ webcam: Mô hình sẽ nhận hình ảnh từ webcam, xử lý qua YOLO11 để nhận diện các cử chỉ tay và phân loại các ký hiệu SL.
* Nhận diện và hiển thị kết quả: Mỗi khung hình video từ webcam sẽ được xử lý để nhận diện các cử chỉ tay, đồng thời các ký hiệu sẽ được hiển thị trực quan trên giao diện người dùng.

## **. Cài đặt giao diện người dùng với Streamlit**

Streamlit được sử dụng để xây dựng giao diện người dùng đơn giản và hiệu quả, giúp hiển thị kết quả nhận diện từ webcam.

* Hiển thị webcam: Streamlit giúp hiển thị hình ảnh từ webcam, cho phép người dùng dễ dàng xem kết quả nhận diện trong thời gian thực.
* Kết quả nhận diện: Sau khi YOLO11 nhận diện cử chỉ tay, kết quả được hiển thị trên giao diện người dùng, bao gồm các ký hiệu và hộp bao quanh đối tượng nhận diện.

Giao diện này giúp người dùng dễ dàng tương tác với hệ thống mà không gặp khó khăn, đồng thời cung cấp kết quả nhận diện chính xác và rõ ràng.

## **. Kết chương**

Chương 3 đã trình bày quy trình thực hiện mô hình nhận diện ngôn ngữ ký hiệu SL sử dụng YOLO11. Các bước từ chuẩn bị dữ liệu, huấn luyện mô hình, triển khai mô hình trong nhận diện thời gian thực, đến việc cài đặt giao diện người dùng với Streamlit đã được thực hiện. Việc triển khai mô hình YOLO11 cùng với giao diện người dùng Streamlit giúp tạo ra một hệ thống nhận diện ngôn ngữ ký hiệu hiệu quả và dễ sử dụng.

Sau đó, chúng ta cùng đến phần đánh giá các kết quả sau khi chạy mô hình và được các kết quả thực tiển.

# **CHƯƠNG 4: KẾT QUẢ VÀ ĐÁNH GIÁ**

## **. Kết quả nhận diện SL từ Webcam**

Trong quá trình triển khai mô hình YOLO11 cho nhận diện ngôn ngữ ký hiệu SL( Sign Language ) qua webcam, kết quả nhận diện được kiểm tra trực tiếp trong môi trường thực tế. Mô hình nhận diện các cử chỉ tay trong SL với sự hỗ trợ của OpenCV và Streamlit, cho phép hiển thị kết quả trong thời gian thực. Các cử chỉ tay như "I Love You",, "Thank You", "Hello", và nhiều ký hiệu khác được nhận diện chính xác và hiển thị trên giao diện người dùng.

* Hiển thị kết quả: Mỗi khi người dùng thực hiện cử chỉ tay, YOLO11 sẽ xác định và phân loại cử chỉ đó, sau đó hiển thị ký hiệu nhận diện được cùng với hộp bao quanh đối tượng. Kết quả được cập nhật trực tiếp và hiển thị trên giao diện Streamlit.
* Tính tương tác: Hệ thống nhận diện và phân loại các cử chỉ tay trong thời gian thực, giúp người khiếm thính giao tiếp hiệu quả với những người không hiểu ngôn ngữ ký hiệu.

Trong quá trình triển khai mô hình YOLO11 để nhận diện ngôn ngữ ký hiệu **SL** từ webcam, chúng ta đã thu được những kết quả rất khả quan. Mô hình nhận diện các cử chỉ tay chính xác và nhanh chóng, giúp người khiếm thính giao tiếp hiệu quả hơn. Các hình ảnh minh họa dưới đây thể hiện kết quả nhận diện trong thời gian thực.

A screenshot of a graph

AI-generated content may be incorrect.

Hình 4. Hình ảnh Confusion Matrix.

**Công thức:**

* Accuracy (Độ chính xác):
  + Accuracy=
  + Tính độ chính xác bằng cách chia số mẫu phân loại đúng cho tổng số mẫu.
* Precision (Độ chính xác):
  + Precision=
  + Precision đo lường tỉ lệ các mẫu dự đoán đúng trong tổng số mẫu được mô hình phân loại là lớp đó.
* Kết quả Confusion Matrix cho thấy mức độ chính xác của mô hình trong việc phân loại các ký hiệu. Các giá trị trong ma trận cho thấy các cử chỉ tay của từng ký hiệu SL được nhận diện đúng và sai.
* Từ ma trận nhầm lẫn, ta có thể thấy mô hình đã phân loại chính xác tất cả các mẫu trong từng lớp, với giá trị "5" xuất hiện ở tất cả các vị trí chéo của ma trận, cho thấy không có lỗi phân loại (hoàn toàn chính xác).

A graph with blue squares

AI-generated content may be incorrect.

Hình 5. Hình ảnh Confusion Matrix Normalized.

* Confusion Matrix Normalized cung cấp cái nhìn chi tiết hơn về tỷ lệ chính xác và sai sót của mô hình qua từng lớp ký hiệu. Các giá trị trong ma trận này phản ánh tỷ lệ chính xác khi mô hình dự đoán đúng mỗi ký hiệu SL.
* Đây là ma trận nhầm lẫn đã chuẩn hóa, cho thấy tỷ lệ chính xác của mỗi lớp. Tất cả các giá trị chéo đều là 1.00, điều này có nghĩa là mô hình đạt được tỷ lệ chính xác 100% đối với từng lớp ký hiệu ASL, chứng tỏ mô hình hoạt động rất tốt.

A graph of confidence curves

AI-generated content may be incorrect.

Hình 6.Hình ảnh F1-Confidence Curve.

Công thức tính:

* F1 Score là trung bình điều hòa giữa Precision và Recall:
* F1=2\*
* Precision: Tỉ lệ các mẫu phân loại đúng trong tổng số mẫu mà mô hình dự đoán là đúng.
* Recall: Tỉ lệ các mẫu thực sự thuộc lớp này mà mô hình nhận diện đúng.

Nhận xét biểu đồ:

* Đường cong F1-Confidence cho thấy mối quan hệ giữa độ tin cậy của mô hình (confidence) và điểm F1.
* Đường cong xanh dương đại diện cho tất cả các lớp, với điểm F1 đạt giá trị rất cao (~0.99) khi độ tin cậy đạt 0.75. Điều này cho thấy mô hình có khả năng đạt được sự cân bằng tốt giữa precision và recall khi độ tin cậy cao.
* Mức độ tự tin: Khi mức độ tự tin của mô hình thấp (gần 0), điểm F1 thấp, cho thấy mô hình gặp khó khăn trong việc phân loại đúng cử chỉ tay.

A graph with text on it

AI-generated content may be incorrect.

Hình 7. Hình ảnh Precision-Recall Curve.

Công thức Precision và Recall:

* **Precision**: Đo lường độ chính xác của mô hình khi phân loại một mẫu vào lớp dương tính.
  + Precision=
  + **TP**: True Positives (Số mẫu đúng mà mô hình phân loại vào lớp dương tính)
  + **FP**: False Positives (Số mẫu không thuộc lớp dương tính nhưng lại bị mô hình phân loại vào lớp đó)
* **Recall**: Đo lường tỉ lệ mẫu dương tính thực sự mà mô hình nhận diện đúng.
  + Recall=
  + **FN**: False Negatives (Số mẫu thuộc lớp dương tính nhưng mô hình lại phân loại sai)
* **Precision** được duy trì gần 1.0 cho tất cả các lớp ký hiệu, cho thấy mô hình có độ chính xác ổn định và ít mắc lỗi trong việc dự đoán đúng cử chỉ tay.
* Các lớp ký hiệu được mô hình nhận diện rất chính xác mà không bị giảm độ chính xác khi **Recall** thay đổi.

A graph of a graph with different colored lines

AI-generated content may be incorrect.

Hình 8. Hình ảnh Recall-Confidence Curve.

* **Recall**: Đo lường tỉ lệ mẫu dương tính thực sự mà mô hình nhận diện đúng.
  + Recall=
  + **FN**: False Negatives (Số mẫu thuộc lớp dương tính nhưng mô hình lại phân loại sai)
* **Confidence:**
  + Confidence=P(class = positive)=σ(z)
  + σ(z)=1+e−z1​ là hàm sigmoid.
  + zzz là giá trị đầu ra của mô hình (ví dụ: output của mạng neural hoặc điểm số từ mô hình học máy).
  + Nếu **Confidence** gần 1, mô hình rất tự tin vào dự đoán là lớp dương tính.
  + Nếu **Confidence** gần 0, mô hình rất tự tin vào dự đoán là lớp âm tính.
* Các lớp như **Hello**, **I love you**, **No**, **Please**, **Thanks**, và **Yes** đều có **recall** rất cao, gần như đạt tối đa tại các ngưỡng độ tin cậy thấp. Điều này có nghĩa là mô hình rất ít bỏ sót các cử chỉ, đặc biệt là khi độ tin cậy thấp.
* Biểu đồ cho thấy **recall** duy trì gần 1.0 khi **confidence** thấp (dưới 0.8). Điều này chỉ ra rằng mô hình có thể phát hiện hầu hết các cử chỉ ngay cả khi độ tin cậy thấp, đảm bảo không bỏ sót các mẫu thực tế quan trọng.
* Đường **"all classes"** thể hiện **recall** chung cho tất cả các lớp. Dường như mô hình duy trì **recall** gần như 1.0 khi **confidence** thấp và giảm dần khi **confidence** tăng lên. Điều này cho thấy mô hình có khả năng nhận diện chính xác cao ở những ngưỡng tin cậy thấp nhưng có thể bỏ sót khi độ tin cậy quá cao.

A graph of a train and smooth

AI-generated content may be incorrect.

Hình 9. Hình ảnh Training Loss.

Nghĩa của biểu đồ:

* Trục X: số epoch
* Trục Y: số Y

Công thức box\_loss(tổn thất hộp giới hạn):

* Box Loss = 1 – CloU(B,)]
* **B** là hộp giới hạn thực tế (ground truth).
* là hộp giới hạn dự đoán.
* **CloU (Complete IoU)** cải tiến so với IoU (Intersection over Union) bằng cách bổ sung thông tin về khoảng cách giữa hộp và tỷ lệ khung hình.

Công thức cls\_loss:

Sử dụng **Cross-Entropy Loss** hoặc **Binary Cross-Entropy (BCE) Loss** cho bài toán phân loại nhị phân:

Class Loss=−*∑\_i y\_i log(ŷ\_i) + (1 - y\_i) log(1 - ŷ\_i)*

* Trong đó:  
  y\_i là nhãn thực tế của lớp.  
  ŷ\_i là xác suất dự đoán của lớp.  
    
  Ngoài ra, có thể sử dụng Focal Loss để cải thiện phân loại các đối tượng khó, công thức trở thành:  
    
  Class Loss = - α(1 - ŷ\_i)^γ y\_i log(ŷ\_i)  
    
  Trong đó:  
  - α là hệ số điều chỉnh tỉ lệ giữa các lớp.  
  - γ là hệ số siêu tham số để giảm ảnh hưởng của các mẫu dễ phân loại.

Trong đó:

* yiy\_iyi​ là nhãn thực tế của lớp.
* y^i\hat{y}\_iy^​i​ là xác suất dự đoán của lớp.

**train/box\_loss:**

* **Box Loss** giảm mạnh trong giai đoạn đầu của quá trình huấn luyện, từ giá trị lớn (~2) xuống mức thấp (~0.4). Điều này cho thấy mô hình nhanh chóng học cách xác định các đối tượng và điều chỉnh vị trí của hộp giới hạn (bounding box).

**train/cls\_loss:**

* **Classification Loss** ban đầu giảm khá nhanh từ giá trị ~5 xuống mức thấp (~0.5). Điều này chứng tỏ mô hình dần học được cách phân loại chính xác các đối tượng (cử chỉ tay trong ngữ cảnh SL).

**train/dfl\_loss:**

* **DFL Loss** cũng có xu hướng giảm mạnh trong giai đoạn đầu, cho thấy mô hình dần tối ưu hóa các quyết định phân loại sâu và chính xác hơn.
* Tuy nhiên, **"DFL Loss"** có vẻ không giảm mạnh như các loại loss khác, điều này có thể cho thấy mô hình vẫn đang cải thiện khả năng phân tích đặc trưng phức tạp của đối tượng.

A graph of a loss

AI-generated content may be incorrect.

Hình 10. Hình ảnh Validation Loss.

**1. Box Loss (val/box\_loss)**

Box loss thường được sử dụng trong các mô hình phát hiện đối tượng để đo độ chính xác của vị trí và kích thước của hộp giới hạn (bounding box) dự đoán so với hộp giới hạn thực tế. Công thức tính box loss có thể bao gồm hai phần chính:  
  
Box Loss = IoU Loss + Localization Loss  
  
- \*\*IoU Loss (Intersection over Union)\*\*: Đo lường sự tương đồng giữa hai hộp giới hạn (hộp giới hạn dự đoán và hộp giới hạn thực tế).  
 IoU được tính theo công thức:  
 IoU = (Area of intersection) / (Area of union)  
  
- \*\*Localization Loss\*\*: Đo lường độ sai lệch trong các giá trị vị trí của hộp giới hạn, như tọa độ trung tâm của hộp và các kích thước chiều dài, chiều rộng.

**2. Classification Loss (val/cls\_loss)**

Classification loss (hay còn gọi là \*\*cls\_loss\*\*) được sử dụng trong các bài toán phân loại đối tượng. Đây thường là \*\*Cross-Entropy Loss\*\*, tính toán sự sai khác giữa phân phối xác suất dự đoán và phân phối xác suất thực tế (ground truth) của các lớp.  
  
Công thức tính Classification Loss:  
  
Classification Loss = - ∑\_i y\_i log(ŷ\_i)  
  
Trong đó:  
- \*\*y\_i\*\* là nhãn thực tế của lớp.  
- \*\*ŷ\_i\*\* là xác suất dự đoán của lớp \(i\).

**3. Distillation Loss (val/dfl\_loss)**

Distillation Loss (hay \*\*DFL\*\*) thường được dùng trong các mô hình học sâu để truyền đạt thông tin giữa các mạng con hoặc từ mô hình thầy sang mô hình trò. Công thức cho distillation loss có thể như sau:  
  
Distillation Loss = α · KL Divergence + β · MSE Loss  
  
- \*\*KL Divergence\*\* (Kullback-Leibler Divergence): Đo lường sự khác biệt giữa hai phân phối xác suất.  
- \*\*MSE Loss (Mean Squared Error)\*\*: Đo lường sự sai khác giữa các giá trị dự đoán và giá trị thực tế.

* val/box\_loss: **Box Loss** bắt đầu giảm nhanh chóng từ khoảng 2.0 xuống dưới 1.0 trong những giai đoạn đầu. Điều này cho thấy mô hình có khả năng chính xác trong việc xác định các đối tượng và điều chỉnh các hộp giới hạn (bounding boxes) nhanh chóng.
* val/cls\_loss: **Classification Loss** cũng giảm nhanh từ giá trị cao (~4.0) xuống mức thấp (~1.0). Điều này chứng tỏ mô hình đang học tốt các đặc trưng của từng lớp và cải thiện khả năng phân loại đối tượng qua từng epoch kiểm tra.
* val/dfl\_loss: **DFL Loss** có sự giảm rõ rệt, từ khoảng 2.5 xuống mức gần 1.0 trong suốt quá trình kiểm tra. Điều này cho thấy mô hình không chỉ học cách phân loại mà còn tối ưu hóa các quyết định phân tích sâu hơn đối với các đối tượng.

A graph of a number of graphs

AI-generated content may be incorrect.

Hình 11. Hình ảnh Metrics.

* **Precision** thể hiện tỷ lệ dự đoán chính xác so với tổng số dự đoán. Biểu đồ cho thấy **precision** bắt đầu ở mức thấp và nhanh chóng đạt gần 1.0, nghĩa là mô hình đã học được cách phân loại chính xác các đối tượng sau một số epoch ban đầu.
* **Recall** đo lường khả năng của mô hình trong việc phát hiện tất cả các đối tượng. Biểu đồ cho thấy **recall** tăng dần và đạt gần 1.0 khi các epoch tiếp tục. Đặc biệt, khi recall đạt mức cao, mô hình đã nhận diện được hầu hết các đối tượng.
* **mAP50** (Mean Average Precision at IoU threshold 0.50) là chỉ số đo lường độ chính xác trung bình trên tất cả các lớp khi tỷ lệ chồng lấn giữa các hộp giới hạn lớn hơn 50%. Biểu đồ cho thấy **mAP@50** đạt gần 1.0, chứng tỏ mô hình có độ chính xác cao trong việc phân loại các cử chỉ tay.
* **mAP50-95** đo lường độ chính xác trung bình của mô hình với nhiều ngưỡng chồng lấn (IoU từ 0.50 đến 0.95). Biểu đồ cho thấy chỉ số này cũng đạt gần 0.9, cho thấy mô hình có khả năng nhận diện rất tốt ngay cả khi mức độ chồng lấn giữa các hộp giới hạn thay đổi. A group of blue squares

  AI-generated content may be incorrect.

Hình 12. Hình ảnh tương quan.

Đây là một ma trận phân tán (pairplot) hoặc biểu đồ phân tán theo các cặp thuộc tính với các phân phối của từng thuộc tính trên các trục chính. Trong trường hợp này, các thuộc tính là:

* **x**: Tọa độ x của hộp giới hạn.
* **y**: Tọa độ y của hộp giới hạn.–
* **width**: Chiều rộng của hộp giới hạn.
* **height**: Chiều cao của hộp giới hạn.

**Mô tả các phần của biểu đồ:**

* Các histogram dọc theo đường chéo chính (trong các ô như "x", "y", "width", "height") mô tả phân phối của mỗi thuộc tính riêng biệt.
* Các biểu đồ phân tán ngoài đường chéo thể hiện mối quan hệ giữa các cặp thuộc tính. Ví dụ:
  + Mối quan hệ giữa **x** và **y**.
  + Mối quan hệ giữa **width** và **height**.
  + Mối quan hệ giữa **x** và **width**, v.v.
* Mối quan hệ giữa các đặc trưng: Các biểu đồ tán xạ giữa các đặc trưng cho thấy rằng x, y, width, và height đều có mối quan hệ phụ thuộc với nhau trong không gian dữ liệu.
* Các đặc trưng như width và height có sự tương quan khá mạnh, cho thấy rằng khi một trong các giá trị này thay đổi, giá trị còn lại cũng có sự thay đổi tương ứng.
* x và y không có sự tương quan mạnh mẽ, điều này có thể chỉ ra rằng các cử chỉ tay có thể xuất hiện ở các vị trí ngẫu nhiên trong không gian mà không có sự phụ thuộc chặt chẽ.

**Hình ảnh giao diện streamlit:**

A screenshot of a computer

AI-generated content may be incorrect.

Hình 13. Hình ảnh giao diện streamlit khi thực realtime.

## **. Đánh giá hiệu suất mô hình**

**Ưu Điểm:**

* Hiệu Suất Cao trong Thời Gian Thực:
  + YOLO11 tiếp tục duy trì lợi thế của các phiên bản YOLO trước đó về tốc độ nhận diện thời gian thực. Mô hình này có thể nhận diện các đối tượng, bao gồm các cử chỉ tay trong ngôn ngữ ký hiệu SL, với độ trễ cực kỳ thấp, đáp ứng tốt trong môi trường yêu cầu phản hồi nhanh.
* Độ Chính Xác Cao:
  + Một trong những ưu điểm nổi bật của YOLO11 là độ chính xác cực cao, với khả năng nhận diện và phân loại các cử chỉ tay trong ngữ cảnh SL. Các chỉ số như Precision, Recall, và F1-score của YOLO11 đều đạt được mức rất cao, đặc biệt là trong các điều kiện ánh sáng và nền phức tạp.
  + Mô hình cũng đạt điểm mAP (mean Average Precision) cao, cho thấy khả năng chính xác của mô hình khi nhận diện các đối tượng ở nhiều mức độ chồng lấn (IoU).
* Khả Năng Xử Lý Các Đối Tượng Phức Tạp:
  + YOLO11 có khả năng nhận diện các đối tượng nhỏ và phức tạp rất tốt. Đặc biệt khi nhận diện các cử chỉ tay với các đối tượng có hình dạng phức tạp hoặc các đối tượng trong môi trường đông đúc.
* Cải Thiện Trong Việc Nhận Diện Các Đối Tượng Nhỏ:
  + Một điểm nổi bật của YOLO11 là khả năng nhận diện các đối tượng có kích thước nhỏ, điều này cực kỳ quan trọng trong việc nhận diện các ký hiệu SL mà có thể xuất hiện với kích thước nhỏ trên ảnh.
* Khả Năng Tối Ưu Hóa Tốt Hơn:
  + YOLO11 có các thuật toán tối ưu hóa tiên tiến giúp mô hình có thể hoạt động mượt mà hơn trong môi trường thực tế, với khả năng xử lý dữ liệu nhanh và chính xác mà không cần yêu cầu phần cứng quá mạnh mẽ như các phiên bản trước.

**Nhược Điểm:**

* Cần Phần Cứng Mạnh Mẽ:
  + Mặc dù YOLO11 đã được tối ưu hóa để hoạt động hiệu quả hơn, phần cứng mạnh mẽ (đặc biệt là GPU với hiệu suất cao) vẫn là yêu cầu tối thiểu để đạt được hiệu suất tối ưu. Điều này có thể tạo ra rào cản đối với việc triển khai mô hình trên các thiết bị di động hoặc các hệ thống có phần cứng hạn chế.
* Phải Được Huấn Luyện Trên Dữ Liệu Đầy Đủ và Đa Dạng:
  + YOLO11, như các phiên bản trước, vẫn cần một bộ dữ liệu huấn luyện đầy đủ và đa dạng để đạt được kết quả tốt nhất. Nếu bộ dữ liệu huấn luyện quá nhỏ hoặc không đại diện cho các tình huống thực tế, mô hình có thể gặp phải tình trạng overfitting hoặc không nhận diện chính xác các cử chỉ tay ít gặp hoặc phức tạp.
* Tính Toán Cao:
  + Mặc dù tốc độ nhận diện của YOLO11 rất nhanh, nhưng việc huấn luyện mô hình và tính toán trong thời gian thực yêu cầu tài nguyên tính toán lớn. Điều này có thể làm giảm khả năng triển khai mô hình trên các nền tảng có tài nguyên tính toán hạn chế.
* Khó Khăn Khi Nhận Diện Các Cử Chỉ Tương Tự:
  + Mặc dù YOLO11 có thể nhận diện cử chỉ tay chính xác, nhưng mô hình vẫn có thể gặp khó khăn khi nhận diện các cử chỉ tay tương tự nhau trong ngôn ngữ ký hiệu SL. Các ký hiệu có hình dạng gần giống nhau như "T", "U", hoặc "V" có thể gây nhầm lẫn trong quá trình nhận diện, đặc biệt trong môi trường có nhiễu hoặc ánh sáng không đồng đều.
* Cải Tiến Cần Thiết Cho Điều Kiện Môi Trường Khó Khăn:
  + Mặc dù YOLO11 có khả năng nhận diện tốt trong môi trường chuẩn, nhưng khi đối mặt với nền phức tạp hoặc điều kiện ánh sáng yếu, độ chính xác của mô hình có thể giảm. Việc tối ưu hóa mô hình cho các điều kiện này vẫn là một thử thách cần giải quyết.

## **. Kết chương**

Trong chương này, chúng ta đã thực hiện đánh giá chi tiết về hiệu suất của mô hình YOLO11 trong việc nhận diện ngôn ngữ ký hiệu SL qua webcam. Các kết quả thu được từ việc huấn luyện, kiểm tra, và đánh giá mô hình đã chứng minh rằng YOLO11 là một công cụ mạnh mẽ trong việc nhận diện các cử chỉ tay trong thời gian thực.

YOLO11 đã chứng minh được khả năng mạnh mẽ trong việc nhận diện ngôn ngữ ký hiệu SL, đặc biệt là trong môi trường thời gian thực và với các ký hiệu phổ biến. Tuy nhiên, vẫn còn một số thách thức cần giải quyết để tối ưu hóa mô hình cho các điều kiện thực tế phức tạp. Các bước tiếp theo sẽ tập trung vào việc tăng cường dữ liệu huấn luyện và tối ưu hóa mô hình để nâng cao độ chính xác và tính ổn định khi triển khai trong các tình huống thực tế.

# **CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

## **. Kết luận**

Trong khuôn khổ nghiên cứu này, mô hình YOLO11 đã được áp dụng để giải quyết bài toán nhận diện ngôn ngữ ký hiệu SL( Sign Language ). Quá trình nghiên cứu và thử nghiệm đã chỉ ra rằng YOLO11 là một công cụ mạnh mẽ và hiệu quả trong việc nhận diện các cử chỉ tay trong ngữ cảnh SL với độ chính xác và tốc độ xử lý ấn tượng.

Các kết quả đạt được từ các chỉ số như Precision, Recall, F1-score, và mAP cho thấy YOLO11 có khả năng phân loại chính xác các ký hiệu SL phổ biến, đặc biệt là khi môi trường ánh sáng đủ và nền không quá phức tạp. Mô hình cũng có khả năng xử lý các đối tượng nhỏ và phức tạp trong ảnh, giúp nhận diện các ký hiệu SL rõ ràng và nhanh chóng.

Tuy nhiên, mô hình vẫn còn một số hạn chế, đặc biệt là trong các điều kiện môi trường thay đổi như ánh sáng yếu, nền phức tạp, hoặc khi nhận diện các cử chỉ tay ít gặp. Những vấn đề này cần được giải quyết để nâng cao hiệu suất của mô hình trong môi trường thực tế.

Với những kết quả đạt được, YOLO11 đã chứng minh được giá trị của mình trong việc phát triển hệ thống nhận diện ngôn ngữ ký hiệu SL, mở ra cơ hội cho việc ứng dụng công nghệ vào việc hỗ trợ giao tiếp cho người khiếm thính.

## **. Hướng phát triển**

Dựa trên các kết quả và phân tích trong nghiên cứu này, các hướng phát triển tiếp theo có thể bao gồm:

* Tăng Cường Dữ Liệu Huấn Luyện:
  + Một trong những yếu tố quan trọng trong việc cải thiện mô hình là tăng cường bộ dữ liệu huấn luyện. Việc thu thập và bổ sung thêm các cử chỉ tay ít gặp, cũng như các cử chỉ tay trong môi trường ánh sáng yếu và nền phức tạp, sẽ giúp mô hình cải thiện độ chính xác và khả năng tổng quát của mình.
  + Augmentation dữ liệu (tăng cường dữ liệu) có thể được áp dụng để làm đa dạng bộ dữ liệu, giúp mô hình học được nhiều tình huống hơn và giảm thiểu overfitting.
* Cải Tiến Mô Hình Cho Các Điều Kiện Thực Tế:
  + Mặc dù YOLO11 đã hoạt động tốt trong điều kiện lý tưởng, nhưng cải tiến mô hình để hoạt động tốt hơn trong các điều kiện ánh sáng yếu và nền phức tạp là điều cần thiết. Các kỹ thuật như adaptive thresholding, image segmentation, và denoising có thể được áp dụng để giúp mô hình hoạt động hiệu quả hơn trong các môi trường không ổn định.
* Tối Ưu Hóa Hiệu Suất Mô Hình:
  + Để triển khai mô hình trên các thiết bị có cấu hình phần cứng hạn chế, cần tối ưu hóa mô hình để giảm thiểu yêu cầu tài nguyên tính toán. Việc giảm kích thước mô hình và tối ưu hóa các tham số có thể giúp mô hình chạy mượt mà trên các thiết bị di động và máy tính với phần cứng không mạnh mẽ.
* Phát Triển Các Tính Năng Mới:
  + Bên cạnh việc nhận diện ngôn ngữ ký hiệu, mô hình YOLO11 có thể được phát triển để tích hợp thêm các tính năng khác, như nhận diện cảm xúc, nhận diện các tình huống giao tiếp trong ngữ cảnh thực tế. Điều này sẽ mở rộng khả năng sử dụng của hệ thống trong việc hỗ trợ giao tiếp đa dạng cho người khiếm thính.
* Ứng Dụng Mở Rộng:
  + YOLO11 có thể được mở rộng để ứng dụng vào nhiều lĩnh vực khác, ngoài nhận diện ngôn ngữ ký hiệu SL. Ví dụ, nhận diện hành vi, nhận diện cử chỉ trong các tình huống giao tiếp khác hoặc hỗ trợ tự động trong các lĩnh vực giáo dục và dịch vụ công cộng.
* Tạo Giao Diện Người Dùng Dễ Dàng Hơn:
  + Để làm cho hệ thống trở nên thân thiện và dễ sử dụng hơn cho người khiếm thính, có thể phát triển thêm các giao diện người dùng trực quan, bao gồm việc hiển thị kết quả nhận diện theo dạng văn bản, phát âm thanh hoặc hiển thị thông tin liên quan như từ điển cử chỉ tay.

# **TÀI LIỆU THAM KHẢO**

<https://aws.amazon.com/vi/what-is/computer-vision/?utm_source=chatgpt.com>

[https://vi.wikipedia.org/wiki/Th%E1%BB%8B\_gi%C3%A1c\_m%C3%A1y\_t%C3%ADnh](https://vi.wikipedia.org/wiki/Th%E1%BB%8B_gi%C3%A1c_m%C3%A1y_t%C3%ADnh%20(1))

<https://aws.amazon.com/vi/what-is/computer-vision/?utm_source=chatgpt.com>

<https://pjreddie.com/darknet/yolo/>

<https://pyimagesearch.com/>

<https://docs.opencv.org/master/>

<https://www.sciencedirect.com/science/article/abs/pii/S1877053520300907>

<https://www.igi-global.com/chapter/sign-language-recognition-system-using-deep-learning-models/110569>